

WHAT IS CLAIMED IS:

- 1 1. A method for storing data in a data store, comprising:
2 receiving a first file in a first file format including essence, metadata objects
3 providing information on the essence, and a unique identifier assigned to the essence;
4 extracting the essence from the file;
5 storing the essence in the data store;
6 extracting metadata from each metadata object in the first file; and
7 storing the extracted metadata in the data store in a second file format, wherein
8 the extracted metadata and essence in the data store are accessible using the unique
9 identifier assigned to the essence.
- 1 2. The method of claim 1, wherein storing the extracted metadata in the data
2 store in the second file format further comprises:
3 adding the extracted metadata to a metadata data structure in the second file
4 format, wherein the metadata data structure is stored in the data store.
- 1 3. The method of claim 2, wherein the metadata data structure comprises a
2 metadata file, and wherein one separate metadata file is provided for each received file in
3 the first file format to store in the data store.
- 1 4. The method of claim 2, wherein adding the extracted metadata to the
2 metadata data structure further comprises storing the metadata extracted from each
3 metadata object in the first file in at least one field in the metadata data structure.
- 1 5. The method of claim 4, wherein each metadata object in the first file
2 includes a metadata code corresponding to a metadata type, and wherein storing the
3 extracted metadata from each metadata object in the metadata data structure further
4 comprises:

5 providing a mapping associating field types for the second file format with the
6 metadata codes identifying metadata types in the first file format, and
7 determining from the mapping, for each metadata object, one field type for the
8 metadata code in the metadata object, wherein one field in the metadata data structure
9 including the metadata indicates the determined field type.

1 6. The method of claim 5, wherein storing the metadata in the data structure
2 for each metadata object further comprises storing a metadata value in an additional field
3 in the metadata structure if the metadata object includes one metadata value.

1 7. The method of claim 5, wherein each field in the metadata data structure
2 comprises a tagged field and wherein the metadata data structure is in a tagged file
3 format.

1 8. The method of claim 7, wherein storing the metadata in the at least one
2 field further comprises:
3 generating a tagged element field indicating the determined field type if the
4 metadata object does not include an attribute value;
5 inserting the generated tagged element field into the metadata data structure;
6 generating a tagged attribute field including an attribute value extracted from the
7 metadata object if the metadata object includes the attribute value; and
inserting the generated tagged attribute field into the metadata data structure.

1 9. The method of claim 1, wherein the essence data comprises one of
2 multimedia data, application data, text, and database records.

1 10. The method of claim 9, wherein the first file format comprises the Media
2 Exchange Format (MXF) and wherein the metadata objects in the first file are
3 implemented in the Key, Length, Value (KLV) coding scheme.

1 11. The method of claim 1, further comprising:
2 receiving a unique identifier;
3 accessing the essence and the metadata data structure associated with the unique
4 identifier;
5 generating at least one reconstructed metadata object from the metadata in the
6 metadata data structure; and
7 assembling a second file in the first file format including the reconstructed
8 metadata object, the accessed essence, and the received unique identifier.

1 12. The method of claim 11, wherein the metadata data structure includes
2 metadata in tagged fields, wherein generating each reconstructed metadata object
3 comprises:
4 accessing the metadata data from the tagged fields; and
5 storing the accessed metadata in the reconstructed metadata object.

1 13. The method of claim 12, wherein each metadata object in the first file
2 format includes a metadata code corresponding to a metadata type, and wherein
3 generating each reconstructed metadata object further comprises:
4 providing a mapping associating field types for the second file format with the
5 metadata codes identifying metadata types in the first file format;
6 determining from the accessed metadata data structure one field type;
7 determining from the mapping the metadata code corresponding to the determined
8 field type; and
9 including the determined metadata code in the reconstructed metadata object.

1 14. The method of claim 13, wherein generating the reconstructed metadata
2 object further comprises:
3 accessing a metadata attribute value from the accessed metadata data structure for
4 the accessed field type; and

5 adding the accessed attribute value to the reconstructed metadata object for the
6 field type.

1 15. The method of claim 1, wherein the first and second file formats are a
2 same file format.

1 16. The method of claim 1, wherein the first and second file formats are
2 different file formats.

1 17. A system for storing data, comprising:
2 a data store;
3 means for receiving a first file in a first file format including essence, metadata
4 objects providing information on the essence, and a unique identifier assigned to the
5 essence;
6 means for extracting the essence from the file;
7 means for storing the essence in the data store;
8 means for extracting metadata from each metadata object in the first file; and
9 storing the extracted metadata in the data store in a second file format, wherein
10 the extracted metadata and essence in the data store are accessible using the unique
11 identifier assigned to the essence.

1 18. The system of claim 17, wherein the means for storing the extracted
2 metadata in the data store in the second file format further performs:
3 adding the extracted metadata to a metadata data structure in the second file
4 format, wherein the metadata data structure is stored in the data store.

1 19. The system of claim 18, wherein the metadata data structure comprises a
2 metadata file, and wherein one separate metadata file is provided for each received file in
3 the first file format to store in the data store.

1 20. The system of claim 18, wherein the means for adding the extracted
2 metadata to the metadata data structure further stores the metadata extracted from each
3 metadata object in the first file in at least one field in the metadata data structure.

1 21. The system of claim 20, wherein each metadata object in the first file
2 includes a metadata code corresponding to a metadata type, and wherein the means for
3 storing the extracted metadata from each metadata object in the metadata data structure
4 further performs:

5 providing a mapping associating field types for the second file format with the
6 metadata codes identifying metadata types in the first file format, and

7 determining from the mapping, for each metadata object, one field type for the
8 metadata code in the metadata object, wherein one field in the metadata data structure
9 including the metadata indicates the determined field type.

1 22. The system of claim 21, wherein the means for storing the metadata in the
2 data structure for each metadata object further stores a metadata value in an additional
3 field in the metadata structure if the metadata object includes one metadata value.

1 23. The system of claim 21, wherein each field in the metadata data structure
2 comprises a tagged field and wherein the metadata data structure is in a tagged file
3 format.

1 24. The system of claim 23, wherein the means for storing the metadata in the
2 at least one field further performs:

3 generating a tagged element field indicating the determined field type if the
4 metadata object does not include an attribute value;

5 inserting the generated tagged element field into the metadata data structure;

6 generating a tagged attribute field including an attribute value extracted from the
7 metadata object if the metadata object includes the attribute value; and
8 inserting the generated tagged attribute field into the metadata data structure.

1 25. The system of claim 17, wherein the essence data comprises one of
2 multimedia data, application data, text, and database records.

1 26. The system of claim 25, wherein the first file format comprises the Media
2 Exchange Format (MXF) and wherein the metadata objects in the first file are
3 implemented in the Key, Length, Value (KLV) coding scheme.

1 27. The system of claim 17, further comprising:
2 means for receiving a unique identifier;
3 means for accessing the essence and the metadata data structure associated with
4 the unique identifier;
5 means for generating at least one reconstructed metadata object from the metadata
6 in the metadata data structure; and
7 means for assembling a second file in the first file format including the
8 reconstructed metadata object, the accessed essence, and the received unique identifier.
9

1 28. The method of claim 27, wherein the metadata data structure includes
2 metadata in tagged fields, wherein the means for generating each reconstructed metadata
3 object further performs:
4 accessing the metadata data from the tagged fields; and
5 storing the accessed metadata in the reconstructed metadata object.

1 29. The system of claim 28, wherein each metadata object in the first file
2 format includes a metadata code corresponding to a metadata type, and wherein the
3 means for generating each reconstructed metadata object further performs:

4 providing a mapping associating field types for the second file format with the
5 metadata codes identifying metadata types in the first file format;;
6 determining from the accessed metadata data structure one field type;
7 determining from the mapping the metadata code corresponding to the determined
8 field type; and
9 including the determined metadata code in the reconstructed metadata object.

1 30. The system of claim 29, wherein the means for generating the
2 reconstructed metadata object further performs:
3 accessing a metadata attribute value from the accessed metadata data structure for
4 the accessed field type; and
5 adding the accessed attribute value to the reconstructed metadata object for the
6 field type.

1 31. The system of claim 17, wherein the first and second file formats are a
2 same file format.

1 32. The system of claim 17, wherein the first and second file formats are
2 different file formats.

1 33. An article of manufacture including code for storing data in a data store,
2 wherein the code is capable of causing operations to be performed comprising:
3 receiving a first file in a first file format including essence, metadata objects
4 providing information on the essence, and a unique identifier assigned to the essence;
5 extracting the essence from the file;
6 storing the essence in the data store;
7 extracting metadata from each metadata object in the first file; and

8 storing the extracted metadata in the data store in a second file format, wherein
9 the extracted metadata and essence in the data store are accessible using the unique
10 identifier assigned to the essence.

1 34. The article of manufacture of claim 33, wherein storing the extracted
2 metadata in the data store in the second file format further comprises:
3 adding the extracted metadata to a metadata data structure in the second file
4 format, wherein the metadata data structure is stored in the data store.

1 35. The article of manufacture of claim 34, wherein the metadata data
2 structure comprises a metadata file, and wherein one separate metadata file is provided
3 for each received file in the first file format to store in the data store.

1 36. The article of manufacture of claim 34, wherein adding the extracted
2 metadata to the metadata data structure further comprises storing the metadata extracted
3 from each metadata object in the first file in at least one field in the metadata data
4 structure.

1 37. The article of manufacture of claim 36, wherein each metadata object in
2 the first file includes a metadata code corresponding to a metadata type, and wherein
3 storing the extracted metadata from each metadata object in the metadata data structure
4 further comprises:
5 providing a mapping associating field types for the second file format with the
6 metadata codes identifying metadata types in the first file format, and
7 determining from the mapping, for each metadata object, one field type for the
8 metadata code in the metadata object, wherein one field in the metadata data structure
9 including the metadata indicates the determined field type.

1 38. The article of manufacture of claim 37, wherein storing the metadata in
2 the data structure for each metadata object further comprises storing a metadata value in
3 an additional field in the metadata structure if the metadata object includes one metadata
4 value.

1 39. The article of manufacture of claim 37, wherein each field in the metadata
2 data structure comprises a tagged field and wherein the metadata data structure is in a
3 tagged file format.

1 40. The article of manufacture of claim 39, wherein storing the metadata in
2 the at least one field further comprises:
3 generating a tagged element field indicating the determined field type if the
4 metadata object does not include an attribute value;
5 inserting the generated tagged element field into the metadata data structure;
6 generating a tagged attribute field including an attribute value extracted from the
7 metadata object if the metadata object includes the attribute value; and
 inserting the generated tagged attribute field into the metadata data structure.

1 41. The article of manufacture of claim 33, wherein the essence data
2 comprises one of multimedia data, application data, text, and database records.

1 42. The article of manufacture of claim 41, wherein the first file format
2 comprises the Media Exchange Format (MXF) and wherein the metadata objects in the
3 first file are implemented in the Key, Length, Value (KLV) coding scheme.

1 43. The article of manufacture of claim 33, further comprising:
2 receiving a unique identifier;
3 accessing the essence and the metadata data structure associated with the unique
4 identifier;

5 generating at least one reconstructed metadata object from the metadata in the
6 metadata data structure; and
7 assembling a second file in the first file format including the reconstructed
8 metadata object, the accessed essence, and the received unique identifier.

1 44. The article of manufacture of claim 43, wherein the metadata data
2 structure includes metadata in tagged fields, wherein generating each reconstructed
3 metadata object comprises:
4 accessing the metadata data from the tagged fields; and
5 storing the accessed metadata in the reconstructed metadata object.

1 45. The article of manufacture of claim 44, wherein each metadata object in
2 the first file format includes a metadata code corresponding to a metadata type, and
3 wherein generating each reconstructed metadata object further comprises:
4 providing a mapping associating field types for the second file format with the
5 metadata codes identifying metadata types in the first file format;
6 determining from the accessed metadata data structure one field type;
7 determining from the mapping the metadata code corresponding to the determined
8 field type; and
9 including the determined metadata code in the reconstructed metadata object.

1 46. The article of manufacture of claim 45, wherein generating the
2 reconstructed metadata object further comprises:
3 accessing a metadata attribute value from the accessed metadata data structure for
4 the accessed field type; and
5 adding the accessed attribute value to the reconstructed metadata object for the
6 field type.

1 48. The method of claim 1, wherein the first and second file formats are
2 different file formats.